AR Drone 2.0 with Subsumption Architecture

Daan Krijnen and Coen Dekker
Artificial Intelligence Research Seminar
Spring 2014
Media Technology MSc Programme
Leiden University, The Netherlands

Abstract

This paper discusses the implementation of a layered control system for a Parrot AR Drone 2.0 based on the subsumption architecture introduced by Rodney Brooks. It includes four layers that subsume one another. Each of the layers functions entirely on its own, based on its own sensor data. The layers consist of flying, object avoidance, face tracking and expression responding.

1. Introduction

In 1988 Herbert the soda can collecting robot was described in an MIT paper my Rodney Brooks, Jonathan Connell and Peter Ning[1]. It was a robot based on the principles of the subsumption architecture as introduced by Brooks [2]. This architecture has caused a shift in thinking about autonomous robots. Unlike most previous autonomous robots, the subsumption architecture divided the tasks of a robot in multiple layers, each having their own goal. This allows robots to respond to their environment in a more natural way, in a parallel processing way. One of the main aspects of the subsumption architecture is that lower layers can overrule higher layers. For instance in the case of Herbert the soda can collecting robot it meant that it would first and foremost just wander around the room and avoid objects. This meant that even if it detects a soda can, the lower layer of object avoidance will prevent it from picking it up if there is a chair in the way.

A few decades later and robots are becoming increasingly more relevant to our society. One of the most interesting developments nowadays is the rising popularity of flying robots, or so called drones. With prices becoming increasingly lower, and computers becoming increasingly faster, it becomes interesting to combine certain fields of computer science of artificial intelligence to analyse the results of these combinations.

In this paper we will discuss the implementation of a subsumption architecture in an autonomous flying robot, a Parrot AR Drone 2.0. We have included layers of face tracking and facial expression classification amongst others in the AR Drone based on the subsumption architecture mentioned earlier.

2. Setup

For this project we used a Parrot AR Drone 2.0 [3]. This quadcopter is controlled via Wifi, and can be programmed by via a SDK. For this project the official SDK has not been used, but and open source library called ofxARDrone has been used with OpenFrameWorks. One of the main aspects of a subsumption architecture is the fact that the inputs and outputs of each layer or module operate separate from one another. An overview of the system setup can be found in figure 1.

The following sensors were available for the layers to make use of:

Accelerometers - These built in sensors give the drone an awareness of its acceleration in each direction.

Gyroscope - The gyroscope provide values of the position it has in space. Whether it leans forward or not.

Movement Sensor - this internal sensor consists of a bottom facing camera. It is used internally in the drone to stabilize the hovering position, and cannot be controlled by us.

Ultrasonic Height Sensor - This sensor provides a

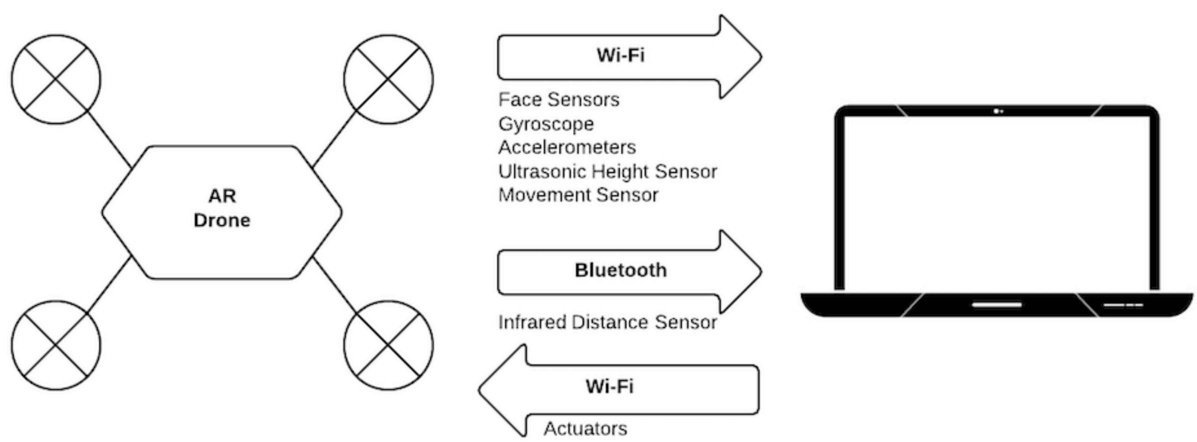

Figure 1: Physical setup

value of the distance between the bottom of the drone and the object below.

Infrared Distance Sensors - These sensors provide the distance between the drone and an object in sight. It has four sensors that we mounted on each side of the drone. They communicate via an Arduino over bluetooth to the main processing machine.

Face Sensor - Using the front facing camera we built a sensor that detects faces in an image. It provides the xy location of the face including the width of the face. It also classifies the expression of the face, whether the face is smiling or the eyebrows are raised. It uses OpenCV for face detection and location tracking, and ofxFaceTracker for the expression classification

The laptop is the operator of the drone and is used for processing of all the data. It is also where we incorporated the subsumption architecture explained in the next sections. Most sensors are built in the AR Drone and can only be accessed via the Wi-Fi connection.

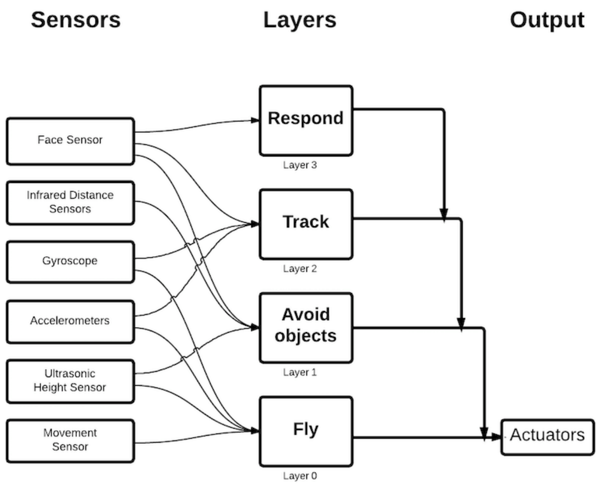

figure 2: Layer overview

3. Layers

The goals of our drone are to find a human face, and interact with it. In order to built this in subsumption architecture as described by Rodney Brooks, we divided this single problem into multiple layers of separate problems that work on top of each other. As the AR Drone has many onboard stabilization algorithms it was not possible for us to control every aspect of the machine. The sensors mentioned in the previous section have been used as input data for the following layers. Figure 2 also shows the layers with their respective sensors. We created four layers of control for our drone:

Layer 0: Fly - The first layer consists of all the commands that are needed to get the drone of the ground into a flying state. Most of the take off is automatically done by the drone which makes this the least controllable layer for us. With this layer turned on the drone only tries to stay at least a certain amount above the ground.

Layer 1 Avoid - Using the infrared sensors this layer develops a sense of space for the drone and calculate whether they are at a safe distance or not. With the use of this data the drone performs a object avoidance maneuver to prevent it from crashing. The objects below the drone make it go up, which is controlled by the internal mechanisms of the AR Drone software, which cannot be altered by us. This layer does not actively steer the drone around the space yet. It merely makes it avoid objects coming into its path.

Layer 2: Track - Using the x/y location of a face provided by the face sensor, this layer adjusts the yaw and pitch movement of the drone in an attempt to keep the face in the center of the image. This causes the drone to follow the face through space. Other built in sensors give the drone an awareness of its own position in space, meaning it knows when a face goes up and down, or if the drone itself goes up and down.

Layer 3: Respond - Working with the data from the face sensor, this layer responds to smiles by performing a manoeuvre that can be described as the wiggling of a tail by a dog. It rolls quickly left and right multiple times, showing the human that is has recognised its smile.

All of the layers above can function in modular way, meaning that they can be switched on or off at will, without influencing the other layers. Obviously, some of these layers such as layer 0 are more essential to the functioning of the system as a whole, but in theory the other layers do not notice a malfunction of layer 0. This also means that more complexity can be added to the project without breaking the layers.

4. Results

The AR Drone functioned most of the time as expected. The added weight of the extra sensors made its flying characteristics less well. Overall the drone did function well in this modular fashion. It takes off, and if all layers are turned on it starts its search sequence for searching a face with the face sensor, at the same time it avoids objects. As soon as it finds a face it will track this face and respond to the expression on that face.

The ability to turn on and off layers is very interesting, it offers the possibility to understand the influences of a single layer on the entire system.

5. Discussion

Even though we did get interesting results with the current configuration, there are some points that we would do different in a future project. To start off with, the original software was not written with the subsumption architecture in mind. This meant that certain variables and functions needed to be rethought and rewritten to fit the layered system. Also the order in which the layers provide their output data to the actuators was an important aspect in achieving a real subsumption architecture.

On a hardware perspective it would be preferable to use lighter sensors and controllers, as a flying robot cannot lift much weight. This would then add better maneuverability and thus tighter responses by the layers.

6. References

[1] R. A. Brooks, J. H. Connell, P. Ning, “Herbert: A Second Generation Mobile Robot”, *MIT AI Memo 1016*, MIT Press, Cambridge, 1988.

[2] R.A. Brooks,, “A Robust Layered Control System For a Mobile Robot”, *IEEE Journal of Robotics and Automation volume 2 issue 1*, IEEE, March 1986, pp. 14-23.

[3] Parrot, http://ardrone2.parrot.com/